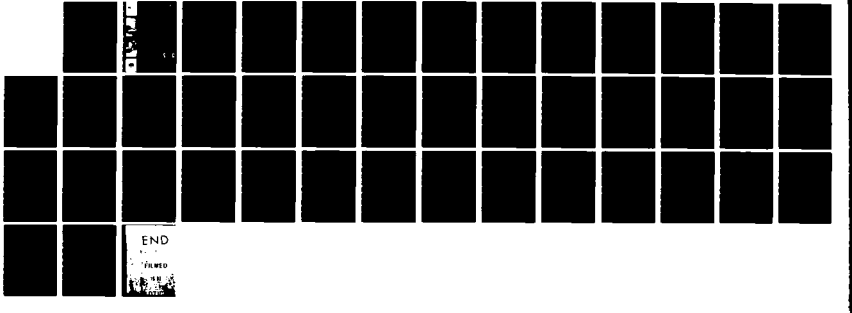
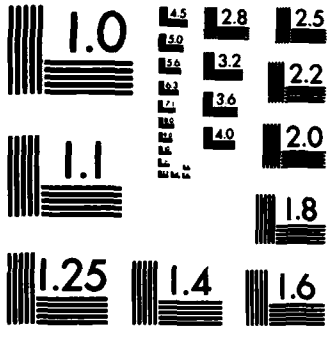
AD-A133 600 COMPUTER-AIDED STRUCTURAL ENGINEERING (CASE) PROJECT USER'S GUIDE: COMPUT.. (U) ARMY ENGINEER WATERWAYS EXPERIMENT STATION VICKSBURG MS J A BREWER JUN 83 1/1

UNCLASSIFIED WES-IR-K-83-2 F/G 9/2 NL





US Army Corps of Engineers

COMPUTER-AIDED STRUCTURAL ENGINEERING (CASE) PROJECT

12

INSTRUCTION REPORT K-83-2

USER'S GUIDE: COMPUTER GRAPHICS PROGRAM FOR GENERATION OF ENGINEERING GEOMETRY (SKETCH)

by

John A. Brewer III

Computer Graphics Research & Applications Laboratory
Mechanical Engineering Department
Louisiana State University, Baton Rouge, La. 70803

June 1983
Final Report

Approved For Public Release; Distribution Unlimited

Prepared for Office, Chief of Engineers, U. S. Army
Washington, D. C. 20314

Monitored by Automatic Data Processing Center
U. S. Army Engineer Waterways Experiment Station
P. O. Box 631, Vicksburg, Miss. 39180

Destroy this report when no longer needed. Do not return it to the originator.

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

The contents of this report are not to be used for advertising, publication, or promotional purposes. Citation of trade names does not constitute an official endorsement or approval of the use of such commercial products.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: Instruction Report K-83-2	2. GOVT ACCESSION NO. AD-A133 600 / 3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): USER'S GUIDE: COMPUTER GRAPHICS PROGRAM FOR GENERATION OF ENGINEERING GEOMETRY (SKETCH)	5. TYPE OF REPORT & PERIOD COVERED: Final report
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): John A. Brewer III	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Computer Graphics Research & Applications Laboratory, Mechanical Engineering Department, Louisiana State University, Baton Rouge, La. 70803	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS: Office, Chief of Engineers, U. S. Army, Washington, D. C. 20314	12. REPORT DATE: June 1983
	13. NUMBER OF PAGES: 37
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): U. S. Army Engineer Waterways Experiment Station, Automatic Data Processing Center, P. O. Box 631, Vicksburg, Miss. 39180	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Available from National Technical Information Service, Springfield, Va. 22151.

This report was prepared under the Computer-Aided Structural Engineering (CASE) Project. A list of published CASE reports is printed on the inside of the back cover.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Computer graphics
Computer programs
Engineering mathematics
Geometry
SKETCH (Computer program)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

SKETCH is a FORTRAN program which is intended to provide quick and painless generation of engineering geometry. The program will utilize any of a number of Tektronix storage tube terminals, including the 4010 and the 4014. Input devices include the terminal keyboard, cross hair thumbwheels, Tek 4953 and 4963 digitizing tablets, and an experimental mouse developed at LSU.

(Continued)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. ABSTRACT (Continued).

The objective of the program is to automatically convert two-dimensional "picture" data into a rich data base consisting of three-dimensional coordinates, lines, polygonal faces, curves, and surfaces. The version of SKETCH (Version 3, Modification 1) that this document reflects does not, as yet, support surface patches (faces with curved boundaries). In order to create complex geometry involving curve segments (and in the future surface patch data), a polyhedra description is roughly sketched, dimensioned, and automatically converted to three-dimensional data. The lines of the precisely dimensioned polyhedra and polygonal cross sections serve as control data and boundary conditions to define arcs, circles, and quadratic or cubic curve forms.

PREFACE

This user's guide documents version 3, modification 1, of SKETCH, a computer program that can be used for generation of engineering geometry. The work in writing the computer program and the user's guide was accomplished with funds provided to the U. S. Army Engineer Waterways Experiment Station (WES), Vicksburg, Miss., by the Civil Works Directorate of the Office, Chief of Engineers, U. S. Army (OCE), under the Computer-Aided Structural Engineering (CASE) Project.

Specifications for the program were provided by the members of the CASE Task Group on 3D Stability. The members of the task group during the period of develpment (though not all served the entire period) were:

Mr. Charles W. Kling, Mobile District (Chairman)
Mr. Robert Haavisto, Sacramento District
Mr. John Hoffmeister, Nashville District
Mr. William Holtham, New England Division
Mr. Gerrett Johnson, Seattle District
Mr. Tom Leicht, St. Louis District
Mr. Thomas J. Mudd, St. Louis District
Mr. John Tang, Sacramento District

The program was developed at Louisiana State University (LSU) in the Computer Graphics Research and Applications Laboratory by Dr. John A. Brewer III. This user's guide was written by Dr. Brewer. A number of graduate students contributed to the development and debugging of SKETCH and to refinement of the documentation. Mr. Colin B. Selleck worked on the command interpreter, Mr. David E. Wilbanks coded the in-core data management system, and Mr. Warren N. Waggenspack, Jr., updated the user's guide and developed the merge capability. Mr. Jeff N. Jortner worked on the circular arc, circle, quadratic, and cubic curve capabilities. A mechanical engineering undergraduate, Ms. Christine M. Behrmann, helped with the documentation and assisted in debugging the program.

The work was performed under the direction of Dr. N. Radhakrishnan, Special Technical Assistant, Automatic Data Processing (ADP) Center,

WES. Mr. Fred T. Tracy, Chief, Research and Development Software Group, ADP Center, served as technical point of contact at WES. Mr. Donald R. Dressler, Structures Division, Civil Works Directorate, was the OCE point of contact. Mr. Donald L. Neumann was Chief, ADP Center.

Directors of WES during the preparation and publication of this report were COL N. P. Conover, CE, and COL T. C. Creel, CE. Technical Director was Mr. F. R. Brown.

CONTENTS

CONTENTS

USER'S GUIDE: COMPUTER GRAPHICS PROGRAM FOR GENERATION OF ENGINEERING GEOMETRY (SKETCH)

INTRODUCTION

SKETCH is a FORTRAN program which is intended to provide quick and painless generation of engineering geometry. The program will utilize any of a number of Tektronix storage tube terminals, including the 4010 and the 4014. Input devices include the terminal keyboard, cross hair thumbwheels, Tek 4953 and 4963 digitizing tablets, and an experimental mouse developed at LSU.

The objective of the program is to automatically convert two-dimensional "picture" data into a rich data base consisting of three-dimensional coordinates, lines, polygonal faces, curves, and surfaces. The version of SKETCH (Version 3, Modification 1) that this document reflects does not, as yet, support surface patches (faces with curved boundaries). In order to create complex geometry involving curve segments (and in the future surface patch data), a polyhedra description is roughly sketched, dimensioned, and automatically converted to three-dimensional data. The lines of the precisely dimensioned polyhedra and polygonal cross sections serve as control data and boundary conditions to define arcs, circles, and quadratic or cubic curve forms.

BASIC CONCEPTS AND DEFINITIONS

Coordinate Axes and Principal Planes. A three-dimensional, right-handed orthogonal coordinate system with axes referenced as X, Y, and Z is assumed. The observation transformation used in producing a picture of a previously sketched object or scene is, however, neither orthogonal nor a standard perspective projection. The transformation is fundamentally a shear transformation which allows coordinate axes to be independently oriented to create the desired picture (see Figure 1).

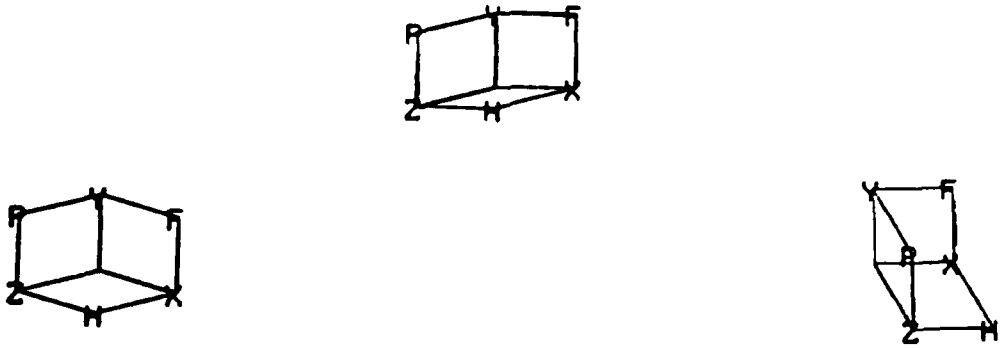


Figure 1. Independent Coordinate Axes for Picture Processing

Each principal plane defined by principal coordinate axes must be arbitrarily named HORIZONTAL, FRONTAL, OR PROFILE. The user is free to associate these names with the principal planes of his choice. The default coordinate system and principal plane naming scheme are shown on the leftmost object in Figure 1.

Line Types. All lines in a rough sketch must be given one of seven possible names by the user. Lines which are parallel to principal axes are designated X, Y, or Z lines. Lines which are not parallel to principal axes, but are parallel to principal planes, are designated H, F, or P lines corresponding to horizontal, frontal, or profile. All other lines are oblique and are named O lines. Figure 2 illustrates each line type.

Object Names. Some user commands optionally reference object names. The object most recently referenced is considered the "active"

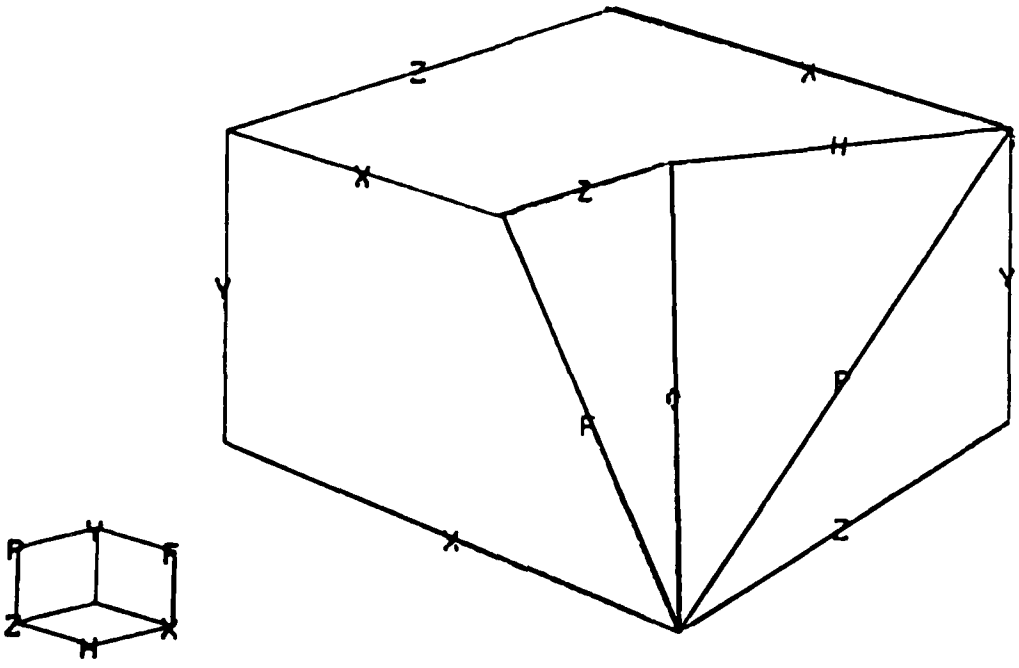


Figure 2. Line Types

object. All commands which do not specifically mention an object name will operate on the currently active object. A new active object may be specified by the USING command (see p. 22).

If only one object is managed at a time, a name is not required. In cases when multiple objects are created in a single session, all objects must carry a unique name consisting of one to six alphanumeric characters beginning with an alpha character. When an unnamed object already exists and a second object is added to the data structure, the unnamed object will automatically be named NONAME by the system. The object NONAME may be renamed with the RENAME command if desired.

There are three words which should not be used as object names: ALL, FACES, AND INTO.

Object Files. Object files are data stored on secondary storage devices (disks). These files can be the result of the SAVE command (see p. 21) or can be created by using the system editor. An example of an object file is given in Figure 3. A single object file may

```
OBJECT    NAME=JAB
COORDINATES
    1          0.000          0.000          0.000
    2          0.000          2.000          0.000
    3          1.000          2.000          0.000
    4          1.000          0.000          0.000
    5          0.000          0.000          3.000
    6          0.000          2.000          3.000
LINES
    1    2     1    Y
    2    2     3    X
    3    4     3    Y
    4    1     4    X
    5    5     1    Z
    6    6     5    Y
    7    2     6    Z
FACES
    1    5     1    2    3    4    1
    2    5     2    1    5    6    2
END
```

Figure 3. An Example Object File

contain multiple objects, but each object should be identified by a unique name. Multiple objects can be merged into a single object by use of the READ command (see p. 13) or the MERGE command (see p. 22).

INPUT PROCEDURES

Command Strings. User commands are processed by two fundamental methods. Word commands or command strings are handled by a general-purpose command interpreter which allows a reasonable margin of user error. For example, errors in spelling beyond the first two, and in some cases three, characters of any word in a command are ignored. Numbers which are part of some commands are interrogated for syntax problems such as multiple decimal points or an imbedded letter inadvertently added in a number. The hyphen character or minus sign is considered an integral part of a number when it is given as the first character of a number. A misplaced hyphen when part of a number generates a syntax error. However, the hyphen may still be used as a part of a word provided it is not the first character of the word. Delimeters between words or numbers can be either blanks or commas.

Graphical Input. The second method of processing user commands is through graphical input. Such input can be achieved by the use of a Tektronix digitizer and stylus or, if a digitizer is not available, by the use of the full-screen cross hairs controlled by thumbwheel potentiometers provided on most Tektronix terminals. An experimental input devise, the "LSU Mouse," is currently under development and is designed to provide an alternative to the digitizer or thumbwheel input hardware. Graphical input is expected as subcommand information after certain word commands are issued. Examples are the SKETCH and DIMENSION commands (see section on Command Syntax and Procedures, p. 11).

When using graphical input to sketch an object, it is not necessary to exercise care in aligning coordinates. Sketches may be constructed by free-hand techniques or digitized from existing drawings. The digitization of either two-dimensional cross sections or three-dimensional pictorials is allowed.

In various instances, graphical input is used to specify a given line, coordinate position, or subcommand by "pointing." Pointing refers to aligning the cross hairs or the lower left corner of the alpha cursor on the desired point or screen location and striking the space bar or

any other keyboard key except <CR>.[1] The cross hairs are moved about using the terminal thumbwheels or the LSU mouse. The alpha cursor follows the pen on the digitizing tablet. Striking a key registers the indicated screen location or point.

[1] It is assumed that the Tektronix terminal is strapped for sending <CR> automatically when any key is depressed while in Graphic Input Mode.

COMMAND SYNTAX AND PROCEDURES

In the following descriptions of command syntax, only those characters which are underlined must be entered to successfully enter a command. Words in parentheses and characters which are not underlined are optional and are completely ignored. Words tabulated in a column format and enclosed by dress brackets ({}) indicate the various alternatives for command words at the indicated position in a command string. Words and characters in straight brackets ([]) are issued by the computer as prompts.

Commands to Redefine or Alter the Coordinate Axes.

CHANGE AXES

The user is allowed to redefine the two coordinate axes for cross sections or the three coordinate axes for three-dimensional pictorials by graphical input methods. The user is also prompted to specify his choices for principal plane names, horizontal, frontal, or profile, in the case of 3D pictorials. An example of using the CHANGE AXES command follows (the numbered descriptions below correspond to the numbered events in Figure 4; note that the circled numbers do not appear on the screen in an actual session):

(1) the user types "CH AXES <CR>",

(2) the computer prompts for information to define the x-axis,

(3) the user enters the first point for defining the x-axis by "pointing" (see definition of "pointing", p. 9),

(5) - (7) same as (2), (3) and (4), respectively, for the y axis,

(8) - (10) same as (2), (3) and (4), respectively, for the z axis,

(11) the computer prompts for plane identification,

(12) the computer identifies a plane (YZ PLANE-, ZX PLANE-, and XY PLANE-) in sequence and the user

types each designation, "H <CR>", "P <CR>", and "F <CR>", respectively,

(13) the new coordinate axes as they appear in the lower left corner of the screen after being redefined,

(14) the default coordinate axis upon entering SKETCH.

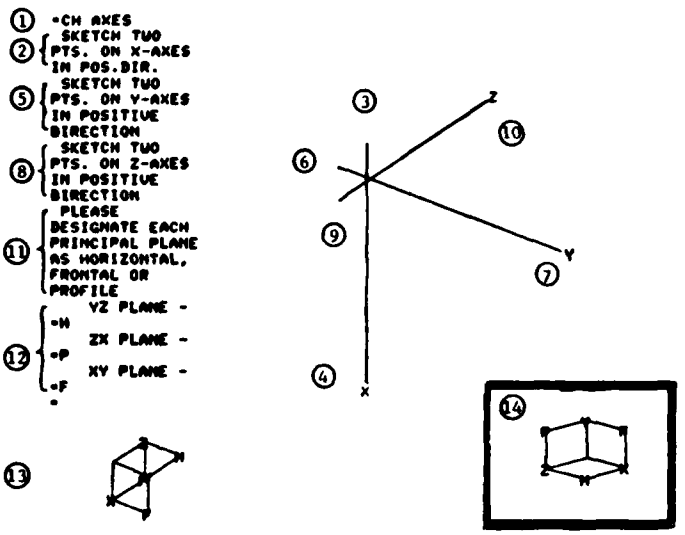


Figure 4. An Example of Using the CHANGE AXIS Command

$$\underline{\text{CHANGE}} \left\{ \begin{matrix} \underline{\text{X}}\text{-AXIS} \\ \underline{\text{Y}}\text{-AXIS} \\ \underline{\text{Z}}\text{-AXIS} \end{matrix} \right\} \quad \text{or} \quad \underline{\text{CHANGE}} \left\{ \begin{matrix} \underline{\text{X}}\ \text{AXIS} \\ \underline{\text{Y}}\ \text{AXIS} \\ \underline{\text{Z}}\ \text{AXIS} \end{matrix} \right\}$$

The user is allowed to reorient an individual axis by graphical input methods.

Data Input and Data Creation Commands.

{INPUT / READ} (objnam)
[FILENAME ?] (filename)

Data which have been previously saved by the SAVE command or created with the system file editor are read into the program. The structure of such data is given in Figure 3 on page 8. The format is free, multiple objects may exist in a single data file, and the END record is optional, as are object names. If an object name is specified with the INPUT command, all data in the stored file are merged into a common object named "objnam." If an object name is not specified, object names contained in the data are retained with individual objects. If names are not found in the data as well, no name is assumed. A name is not required when only one object is involved, but in multiple-object applications, all objects must have a unique name of no more than six (6) characters.

The complete file reference name ("filename") may have up to 25 characters with no embedded blanks or commas.

{SKETCH / MODIFY} (objnam)

This command causes the sketching or digitizing workspace to be outlined and the graphic input menu for subcommands to be drawn (see Figure 5). The terminal is than placed into graphic input mode (cross hairs appear) or the digitizing tablet is armed (alpha cursor appears) if a tablet is being used.

If only a single object is to be currently active, an object name ("objnam") is not required. If a name is not specified and one or more object names are currently in use, the most recently referenced name is assumed for the current sketch.

All available subcommands of the command SKETCH are invoked by "pointing" at menu boxes with a graphic input device (i.e., placing the cross hairs or alpha cursor within a menu box and depressing any keyboard key except <CR>). A selected subcommand is displayed on the left side of the screen in the command space along with a prompt for more information or a description of the appropriate user action to be taken.

POINT

Subcommand POINT permits the establishment of free-standing points which can be used as termination points for lines.

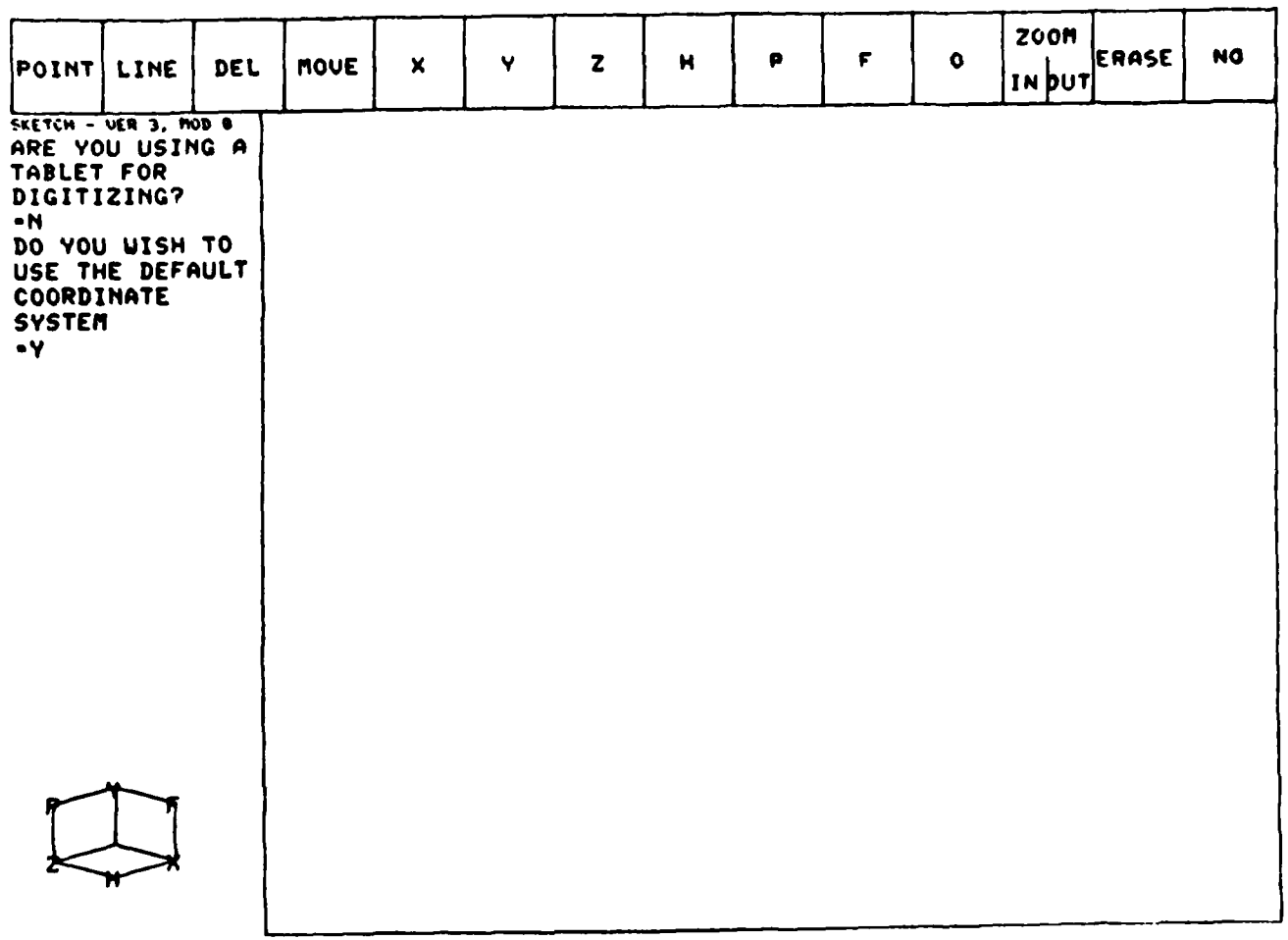


Figure 5. The Sketching Menu and Workspace

However, the POINT command is of little advantage for that purpose since LINE does not require points to be previously established. The primary use for POINT is to designate that points are to be deleted (see the DEL command).

LINE

When LINE is specified the user is prompted to indicate a termination point which may or may not be already established. Subsequent contiguously connected lines may be constructed with no further reference to a menu box. A noncontiguous line can be initiated by pointing to the LINE menu box again. A line which is sketched or digitized more than once is stored in the internal data only once. It is therefore not always necessary for a user to initiate a new line by pointing to the LINE menu box. One may choose to simply overtrace a previously sketched line to reach a terminal point for a new line. It is often possible to sketch complete figures by hitting the LINE menu box only once.

DEL

DEL allows the deletion of points or lines. When the DEL box is indicated, the user is prompted to specify if he intends to delete points or lines. He designates his choice by pointing to the appropriate POINT or LINE menu box. The points or lines to be deleted are also indicated by graphic input (i.e., pointing with the full-screen cross hairs or alpha cursor).

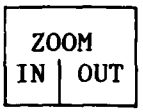
MOVE

MOVE allows points to be repositioned. With the words, "MOVING POINTS," the user is prompted to indicate which point or node is to be repositioned. After pointing to the correct node and depressing any keyboard key other than <CR>, the user is prompted with, "..GOT IT, WHERE TO?" or "-----TRY AGAIN". User response to the first message should be to point to the new coordinate position, in which case all affected lines are immediately redrawn. The latter message indicates that the graphic input cross hairs were not positioned close enough to the desired point.

X	Y	Z	H	P	F	O

The X, Y, Z, H, P, F, and O subcommands are used in designating line types. After pointing to one of these menu boxes,

the user can indicate by graphical input any or all lines to be so named. An incorrectly named line may be renamed at any time.



Pointing to the IN section of the ZOOM menu box results in the prompt, "--ZOOM IN". The user must then indicate, by graphical input, the location of two points in succession which correspond to the lower left and the upper right corners of a new workspace. The screen is then erased and the active data are redisplayed using the newly defined workspace. Pointing to the OUT section of the ZOOM menu box causes the screen to be erased and the active data redisplayed using the original workspace.

The ERASE subcommand clears the entire screen and redisplays all active data.

The NO menu box cannot be activated in Version 3, Modification 1, of the SKETCH program.

The user may exit the SKETCH or MODIFY command mode at any time by pointing to the left of the workspace and below the menu boxes and depressing a keyboard key other than <CR>.

{DIMENSION / REDIMENSION} (objnam)

The dimensioning process is initiated for the indicated object. If an object name is not specified, the most recently referenced name is assumed. If a name has not been specified previously, a name is not required.

This command should be issued only after all lines in a newly created sketch have been named. However, the user need not name lines for a previously dimensioned object. Also, only changes in dimensions are required for objects which are being redimensioned.

The screen is erased, a new menu with only X, Y, and Z boxes is displayed, the object to be dimensioned is redrawn, and extension lines are automatically drawn and labeled XD, YD, or ZD. X, Y, or Z dimensions are set by pointing to pairs of XD, YD, or ZD extension lines, respectively. If the picture is overcrowded with extension lines, the user may request that only one set of extension lines be displayed at a time. This is accomplished by pointing to the appropriate X, Y, or Z menu box. The screen will be erased and the figure redrawn with only the requested extension lines displayed. To display all of the extension lines again, the user must point to any area above the workspace except within the X, Y, or Z boxes and strike some key.

The user is initially asked if some point is to be considered a relative origin. Following a typed "YES" or "Y" response, the user is prompted to indicate the relative origin using graphical input. A "NO" or "N" response will result in a prompt for the user to type in the X, Y, and Z coordinates of a single point in the sketch, and then indicate that point using graphical input.

The user should then "point" to pairs of extension lines and indicate distances by typing when prompted to do so. The first extension line of any pair to be dimensioned should be in a more negative position than the second extension line in a relative sense. There are no rules concerning which dimensions should be given or the order in which they should be specified except that all dimensions must be given from a relatively negative position to a more positive one. If a mistake is made, a dimension may be reestablished at any time. The user is advised when the minimum number of dimensions required in any given direction have been specified and when all dimensions are complete. At that time the user may make any desired corrections by simply respecifying the desired dimension using the procedure outlined above. When he answers with a typed "NO" to the prompt for more corrections, the three-dimensional coordinates for all points are established, and a new picture is calculated but not displayed. The user may issue the DISPLAY command at any time after dimensioning is complete in order to visually inspect the results.

The dimensioning process can be exited at any time by pointing to the left of the workspace and below the menu boxes. This is the normal way to exit when one is redimensioning an object. In the case of a newly created sketch, all dimensioning information is deleted if the dimensioning process is aborted by the user before all dimensions have been completed.

{CENERATE / CREATE} (FACES) (FOR) (objnam)

Face data are automatically generated for completed 3D objects. Line data are not destroyed. Each face (polygon) is displayed one by one and the user is prompted for a decision to retain or reject the face, reverse the order of the nodes, or repeat the display of the face again. The prompt "KEEP?" should be answered by typing "Y" (add this face to the data), "N" (do not add this face to the data), "R" (reverse the node connections for the face), or "A" (display the face again).

CIRCLE {(RADIUS) / (DIAMETER)} (x) {(KEEP) / (DELETE)}

A circle is added to the currently active object. An object may be activated if necessary, with the "USING" command (see p. 22). The object must be completely dimensioned (see DIMENSION command, p. 16) for the CIRCLE command to function. The orientation of a circle is established in the plane of and tangent to any two previously constructed intersecting lines with a common endpoint. The user is prompted to designate the two lines by "pointing" in graphic input mode. The center of the circle will lie on a line which bisects the included angle between the two lines. If the radius or diameter of the circle is not explicitly defined in the command statement, the diameter is assumed to be equal to the three-dimensional length of the first of the two lines designated by the user.

When the DELETE subcommand is given, the intersecting lines will be deleted only if they are of equal length. If the intersecting lines are two sides of an enclosing square, the entire square will be deleted. KEEP ensures that all boundary lines are preserved. RADIUS and KEEP are the default subcommands.

CIRCULAR ARC (RADIUS) x {(KEEP) / (DELETE)}

A circular arc is added to the currently active object. An object may be activated, if necessary, with the USING command (see p. 22). The object must be completely dimensioned (see DIMENSION command, p. 16) for the CIRCULAR ARC command to function. The orientation of a circle is established in the plane of and tangent to any two previously constructed intersecting lines with a common endpoint. The user is prompted to designate the two lines by "pointing" in the graphic input mode. The center of the circular arc will lie on a line which

bisects the included angle between those lines. The radius of the circular arc must be explicitly defined in the command statement.

When the DELETE subcommand is given, the section of each line between its point of tangency with the circular arc and the common endpoint will be deleted. KEEP ensures that all boundary lines are preserved. *RADIUS and KEEP are the default* subcommands.

ELLIPSE (a b) $\left\{\begin{matrix}(\underline{\text{KEEP}}) \\ (\underline{\text{DEL}}\text{ETE})\end{matrix}\right\}$

An ellipse is added to the currently active object. An object may be activated, if necessary, with the USING command (see p. 22). The object must be completely dimensioned (see DIMENSION command, p. 16) for the ELLIPSE command to function. The orientation of an ellipse is established in the plane of and tangent to any two previously constructed intersecting lines with a common endpoint. The user is prompted to designate the two lines by "pointing" in graphic input mode. The dual radii of the ellipse are a and b.

When the DELETE subcommand is given, the intersecting lines are deleted. If the intersecting lines are two sides of an enclosing parallelogram, then all four of the sides of the parallelogram will be deleted. KEEP ensures that all boundary lines are preserved and is the default subcommand.

ELLIPTICAL ARC a b $\left\{\begin{matrix}(\underline{\text{KEEP}}) \\ (\underline{\text{DEL}}\text{ETE})\end{matrix}\right\}$

An elliptical arc is added to the currently active object. An object may be activated, if necessary, with the USING command (see p. 22). The object must be completely dimensioned (see DIMENSION command, p. 16) for the ELLIPTICAL ARC command to function. The orientation of an elliptical arc is established in the plane of and tangent to any two previously constructed intersecting lines with a common endpoint. The user is prompted to designate the two lines by "pointing" in the graphic input mode. The dual radii a and b must both be explicitly defined in the command statement.

When the DELETE subcommand is given, the section of each line between its point of tangency with the elliptical arc and the common endpoint are deleted. KEEP ensures that all boundary lines are preserved and is the default subcommand.

QUADRATIC {(LINES) / (POINTS)} {(KEEP / (DELETE)}

A parametric quadratic curve segment is added to the currently active object. An object may be activated, if necessary, with the USING command (see p. 22). The object must be completely dimensioned (see DIMENSION command, p. 16) for the QUADRATIC command to function. The curve segment is defined by three points. These points can be free-standing or can be the endpoints of any two previously defined intersecting lines having a common endpoint. The user can choose to designate the two intersecting lines or three points by specifying either of the subcommands, LINES or POINTS.

The DELETE subcommand will delete any connecting lines. KEEP ensures that all connecting lines are preserved. LINES and KEEP are the default subcommands.

CUBIC {(LINES) / (POINTS)} {(KEEP) / (DELETE)}

A parametric CUBIC curve segment is added to the currently active object. An object may be activated, if necessary, with the USING command (see p. 22). The object must be completely dimensioned (see DIMENSION command, p. 16) for the CUBIC command to function. The curve segment is defined by four points. These points can be free-standing or can be the endpoints of any three previously defined intersecting lines having two common endpoints. The user can choose to designate the three intersecting lines or four points by specifying either of the subcommands, LINES or POINTS.

The DELETE subcommand will delete any line connecting the second and third points. KEEP ensures that all connecting lines are preserved. LINES and KEEP are the default subcommands.

ARC {(CIRCULAR) / (ELLIPTICAL)} {(RADIUS) a b} x {(KEEP / (DELETE)}

See CIRCULAR ARC and ELLIPTICAL ARC commands (pp. 18 and 19, resp.). CIRCULAR, RADIUS, and KEEP are the default subcommands.

Data Output Commands.

DISPLAY (objnam) (ALL) (NUMBERS)

A display of the indicated object or objects is created on the

CRT screen. The option to display the internal node numbers and current line types is invoked by the entire optional word "NUMBERS."

SAVE (objnam)
[FILENAME ?] (filename)

The SAVE command will create or replace a file named "filename" on secondary storage (disk) with the information contained in the object named "objnam". An object can be saved at any stage of completion.

LIST $\left\{\begin{matrix}\text{(objnm1, objnm2, ...)} \\ \text{(ALL) (OBJECTS)}\end{matrix}\right\}$ (FACES)

A listing of the current numeric values in the internal point and line tables of the indicated object or objects is displayed on the screen. The subcommand OBJECTS results in a complete list of object names. The optional word FACES causes face information to be listed in lieu of line tables. Note: ALL or FACES should not be used as object names.

Utility Commands.

ERASE (objnm1, objnm2, ...) (ALL) (NUMBERS) (LINETYPES)

The screen is erased and the alpha cursor is repositioned at the top left of the command scratch-pad area. If no optional words are specified, the objects which were part of the active display are automatically redisplayed. An object specifically named will not be redrawn. None of the objects are redisplayed if the optional word ALL is given. Note: The word ALL should not be used as an object name. When the NUMBERS or LINETYPES option is specified, the active objects are redisplayed without labeling.

RENAME oldnam (TO) newnam

The object previously referred to as "oldnam" is renamed "newnam".

DELETE objnam

The object named "objnam" is deleted from the system.

SWITCH (TO) $\left\{\begin{matrix}\text{TABLET} \\ \text{CROSS-HAIRS}\end{matrix}\right\}$

Graphical input medium is changed.

CLEAR

The data tables are reinitialized. The current coordinate axes and graphical input media are retained.

RESTART

The program is restarted. All data are cleared.

BYE

The user is prompted to indicate by keyboard input if all necessary data have been saved. A "YES" or "Y" answer terminates the program. Program execution is maintained if "NO" or "N" is typed.

CHARACTER (SIZE) $\left\{\begin{matrix}1\\2\\3\\4\end{matrix}\right\}$

The size of all hardware characters and numbers is changed to the desired size ranging from the smallest, 1, to the largest, 4.

USING objnam

The specified object is subsequently considered the "active" object. Certain commands, such as ARC, CIRCLE, QUADRATIC, and CUBIC, operate on the currently active object.

MERGE objnm1, (objnm2) (INTO) objnm3

The MERGE command combines all coordinate, line, and curve information of "objnm1" and "objnm2," if it is specified, into "objnm3." All redundant information is removed. The object specified to receive the data, "objnm3," may or may not exist.

EXAMPLE SESSIONS

Running the Program. To run SKETCH on the Macon system,* type

```
RUN 11RØKAJAB/SK,R
```

On entering the program, the user is asked to reply to questions concerning the use of a digitizing tablet and the establishment of coordinate axes. Figures 6, 7, and 8 show three typical cases.

A typical digitizing session for the creation of new geometry involves the use of the following command sequence:

```
SKETCH
 .
 .
 (Subcommands and sketching)
 .
 .
DIMENSION
 .
 .
 (Subcommands and dimensional input)
 .
 .
SAVE
 .
 .
 (Computer prompt for filename)
 .
 .
BYE
```

A number of subcommands are involved once the commands SKETCH and DIMENSION are issued, and the use of these subcommands embodies the essence of the SKETCH program.

---

* At present, this program only runs on the Macon system. It will be converted to run on the CDC and Harris computers in due course.

```
SKETCH - VER 3, MOD 0
ARE YOU USING A
TABLET FOR
DIGITIZING?
*NO
DO YOU WISH TO
USE THE DEFAULT
COORDINATE
SYSTEM
*YES
*
```

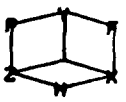


Figure 6. Entering the Program with a Tablet and the Default Coordinate Axes

```
ARE YOU USING A
TABLET FOR
DIGITIZING?
*NO
DO YOU WISH TO
USE THE DEFAULT
COORDINATE
SYSTEM
*NO
WILL YOU BE
DIGITIZING 2-D
CROSS-SECTIONS?
*YES
 THE 2D CROSS
SECTION IS TO
BE
PERPENDICULAR
TO WHICH AXIS?
*Z
 SKETCH TWO
PTS. ON X-AXES
IN POS.DIR.
 SKETCH TWO
PTS. ON Y-AXES
IN POSITIVE
DIRECTION
<CR TO CONTINUE>
```

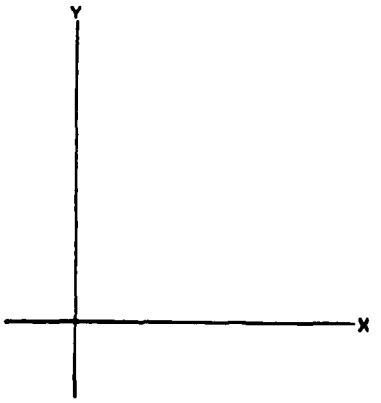


Figure 7. Specifying a Two-Dimensional Reference for Sketching

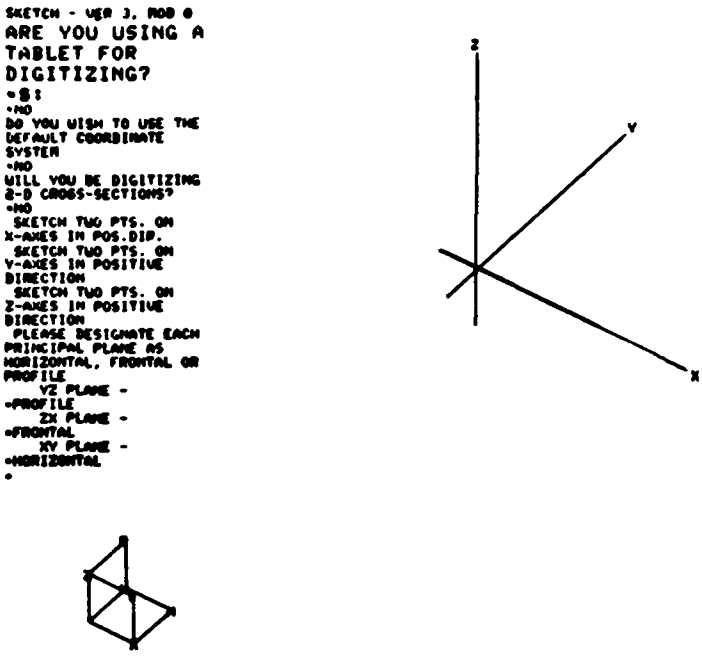


Figure 8. Specifying a Three-Dimensional Reference for Sketching

Sketching A Three-Dimensional Object. As indicated in Figure 9, the user initiates the program and the sketching mode as follows:

(1) the user enters the program using the default coordinate system,

(2) the user enters the sketch mode by typing "SK JUNK",

(3) the subcmmand LINE is invoked by pointing at the corresponding menu box.

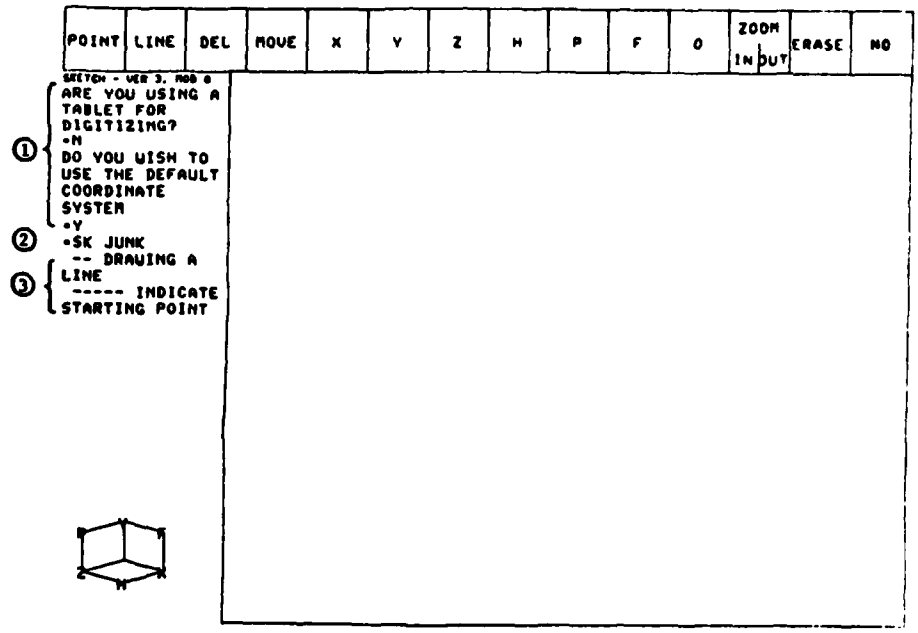


Figure 9. An Example of Initiating a Three-Dimensional (3D) Sketch

Figures 10 and 11 show an initial sketch (still two-dimensional information) in different stages of completion. The lines were drawn by "pointing" at the various locations in the sequence indicated by the numbered circles in the figures. A number of lines were overtraced, eliminating the need to point at the LINE menu box repeatedly. Points and lines are stored internally only once, which means that overtracing does not generate redundant data.

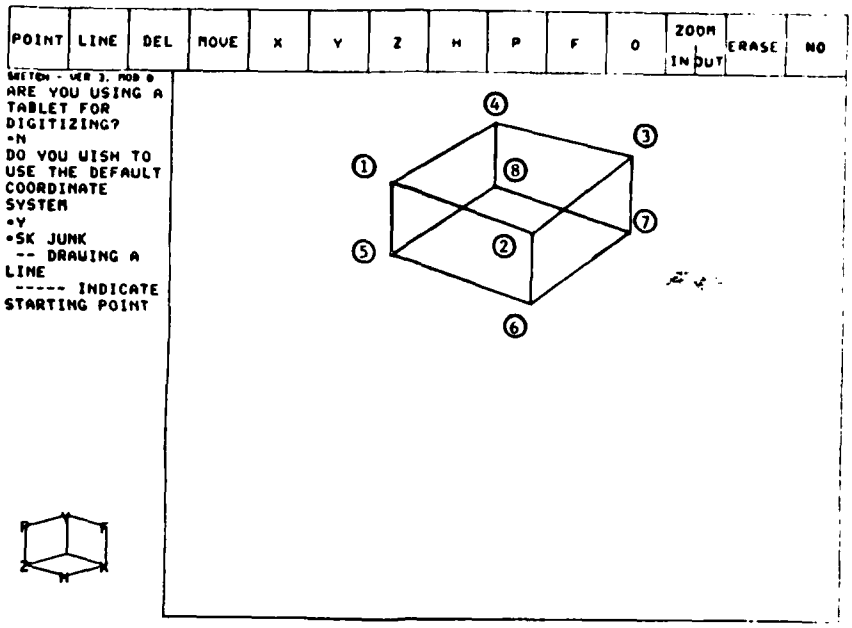


Figure 10. A Partially Complete Sketch of an Object Named "JUNK"

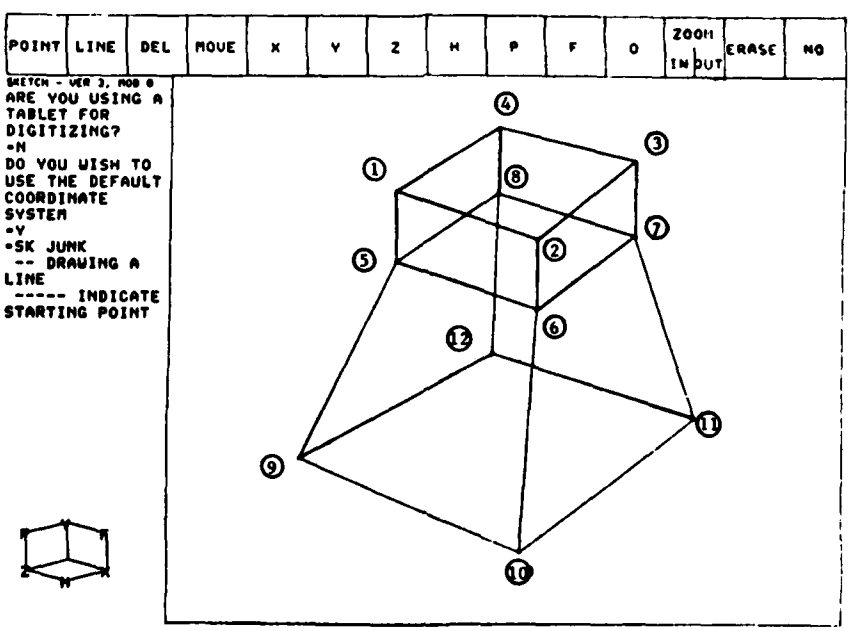


Figure 11. Completing the Topological Definition of the 3D Object

With the sketch in the state shown in Figure 11, pointing to the ERASE menu box results in Figure 12. The command area is cleared, the object and menu area are redisplayed, and the user begins designating line types as indicated below (see Figure 12):

(1) the computer responds to the user pointing at the X menu box,

(2) - (7) the user points to each line parallel to the X axis, and the computer responds by placing an X at the midpoint of each line segment indicated (Note: The order of designating line types is insignificant [X, Y, Z or X, Z, Y or Z, X, Y etc.]),

(8) the computer responds to the user pointing at the Y menu box,

(9) the user utilizes the same procedures as in (2) - (7) to designate each of the four Y lines,

(10) the computer responds to the user pointing at the Z menu box,

(11) the user utilizes the same procedure as in (2) - (7) to designate each of the six Z lines,

(12) the computer responds to the user pointing at the O menu box,

(13) the user utilizes the same procedure as in (2) - (7) to designate each of the four oblique lines,

(14) the user exits the sketching mode by pointing at the command area, and then types "DIMENSION JUNK <CR>". When <CR> is depressed, the screen will be erased and will appear as shown in Figure 13.

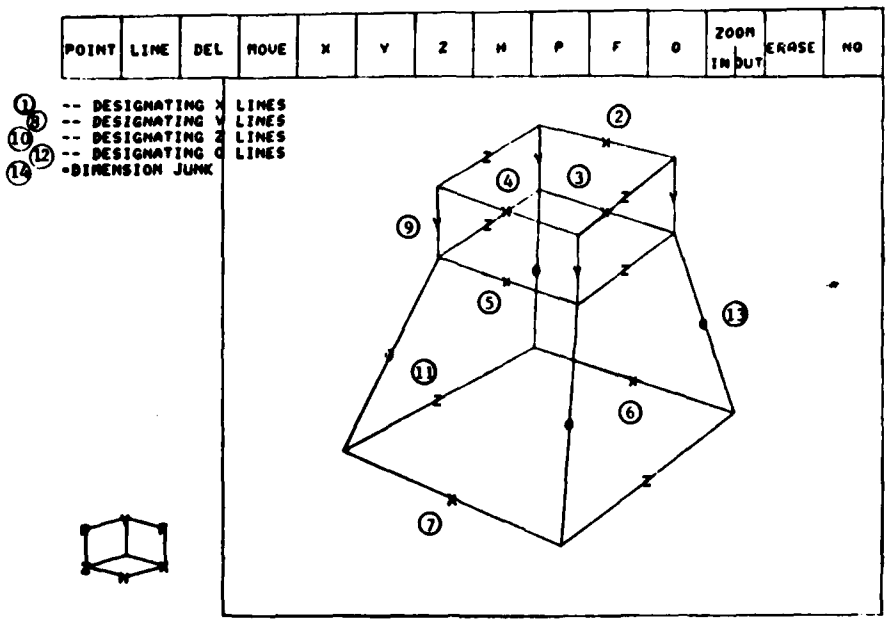


Figure 12. Naming the Lines of a 3D Object

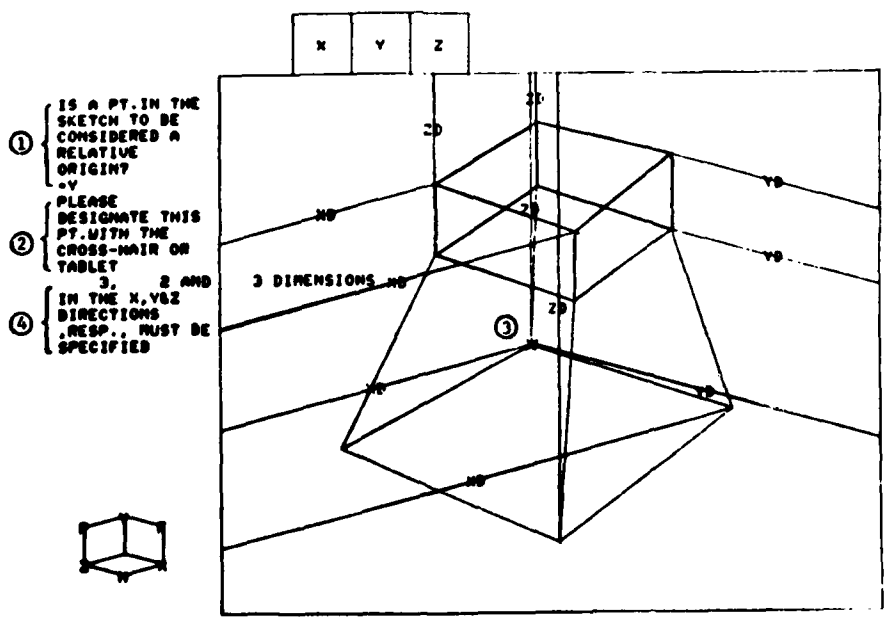


Figure 13. Results of the Command "DIMENSION JUNK"

Figure 13 shows the screen content upon entering the dimensioning mode of the program:

(1) the computer questions the user about a relative origin and the user responds by typing "Y" for yes,

(2) the computer prompts the user to indicate the relative origin,

(3) the user points to the relative origin,

(4) the computer indicates the number of dimensions required in each direction to completely describe the object.

For clarity, the user may request that extension lines be displayed for one dimension only. This is accomplished by pointing to the appropriate menu box (see Figures 14, 15 and 16).

Figure 14 demonstrates the procedure for specifying X dimensions:

(1) - (2) the user points, in sequence, to a pair of extension lines,

(3) the computer prompts the user to type the required distance between the two indicated extension lines (1) - (2),

(4) the user types the correct distance,

(5) - (6) the user points, in sequence, to another set of extension lines,

(7) same as (3) but for extension lines (5) - (6),

(8) same as (4),

(9) - (12) same as (1) - (4) but for a third set of extension lines,

(13) the computer indicates that enough dimensions are specified to define all of the X coordinates.

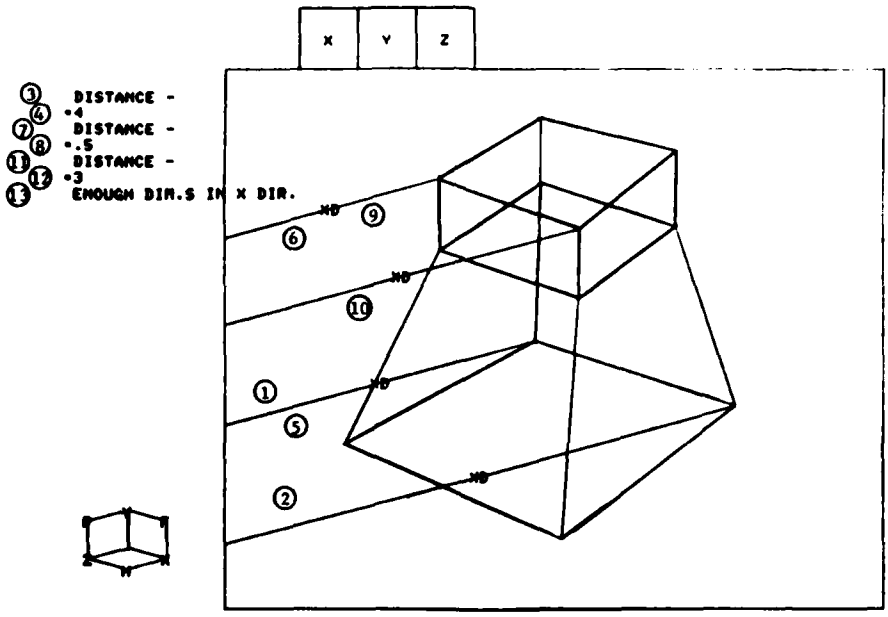


Figure 14. Specifying the X Dimensions

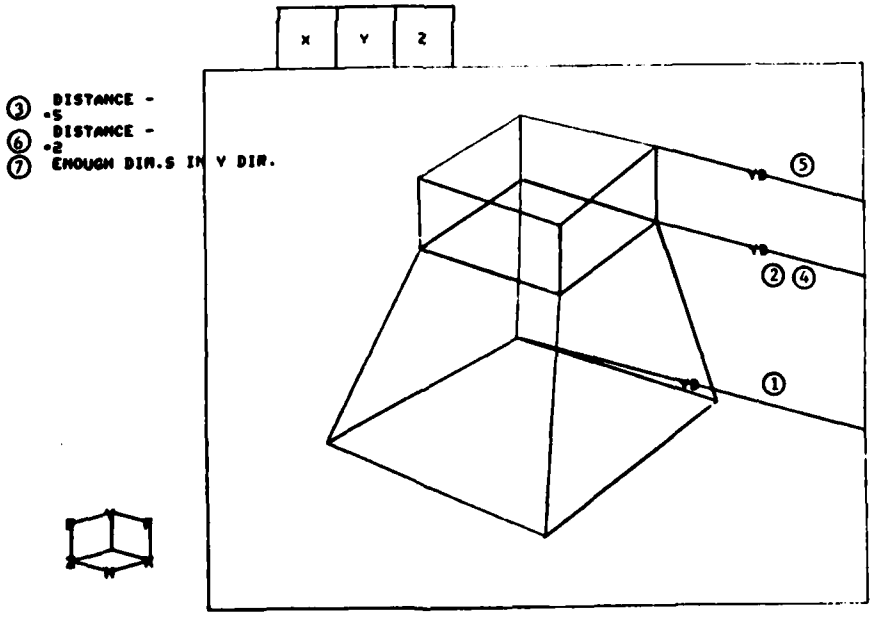


Figure 15. Specifying the Y Dimensions

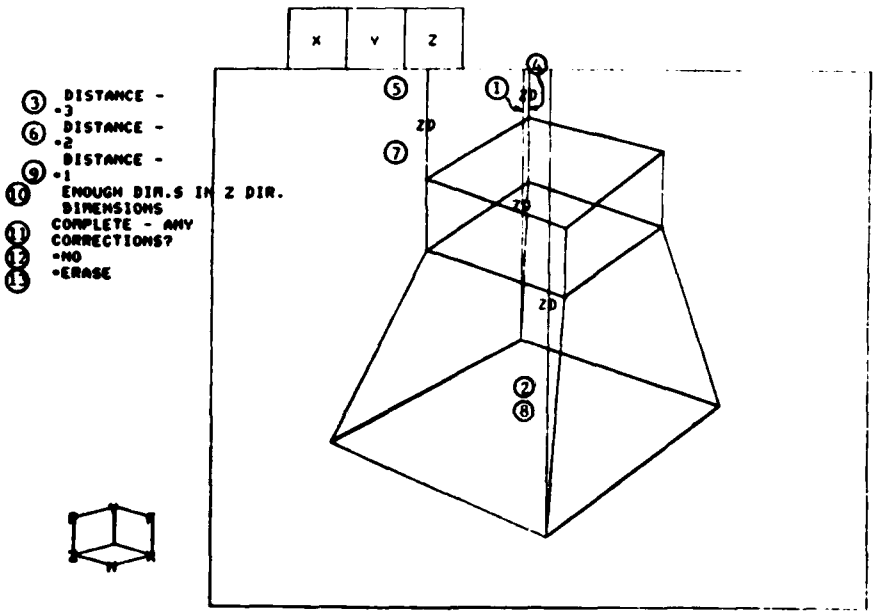


Figure 16. Specifying the Z Dimensions

Figure 15 shows the sequence of events for entering the required dimensions in the Y direction. The procedure is similar to that shown in Figure 14.

Figure 16 shows the sequence of events for entering the required dimensions in the Z direction along with a prompt for corrections:

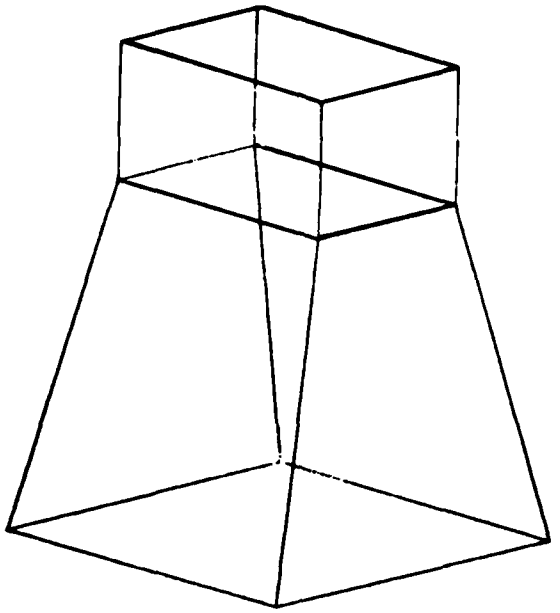
(1) - (10) same as steps (1) - (13) , in Figure 14, except for the Z dimensions.

(11) the computer asks if any corrections are necessary,

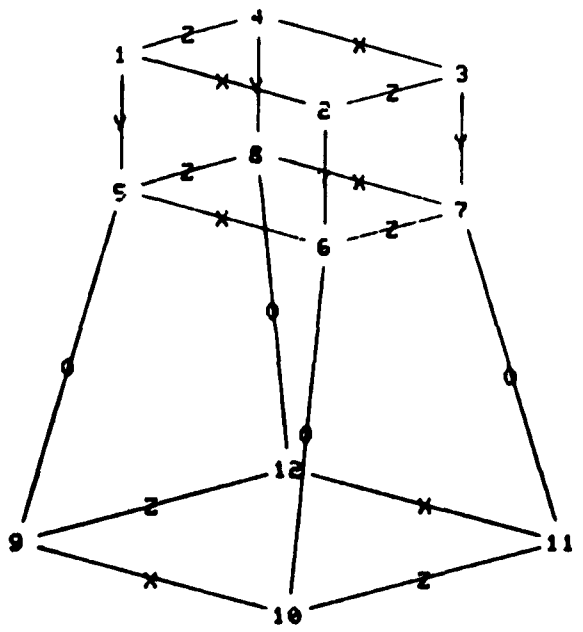
(12) the user types "NO <CR>" since no corrections are required,

(13) the user types "ERASE <CR>". When <CR> is depressed the screen will be erased and the object redisplayed with correct dimensions (see Figure 17a).

At completion of the dimensioning process, the object information is converted to three-dimensional (3D) data. Typing "DISP NUMBERS" results in all node labels and line types being displayed (See Figure 17b).



a.



b.

Figure 17. Completed 3D Object

In order to completely describe an object, the user may then wish to generate face information for the 3D object. Typing "GENERATE FACES" or "GEN" results in the screen being erased and the object being re-displayed with dashed lines as shown in Figure 18. As each face is individually traced with solid lines, the user may keep, reject, reorder, or trace the face again by responding with a typed "Y" or <CR>, "N", "R", or "A" to the computer prompt, "KEEP?". (The numbered descriptions below correspond to the numbered events shown in sequence in Figures 18 through 21):

(1) the computer outlines face (1-2-3-4-1) using solid lines and,

(2) then asks the user whether or not to keep the face,

(3) the user responds with 'Y<CR>' and,

(4) (The insert displays the first face generated),

(5) the computer outlines another face (1-5-6-2-1),

(6) the computer asks whether or not to keep the face,

(7) the user responds with 'Y<CR>',

(8) (The insert displays a clear view of the generated faces),

(9) the computer outlines several more faces in sequence and,

(10) the user responds with 'Y<CR>' (or 'N<CR>' as for face (5-6-7-8-5)),

(11) (The insert displays a clear view of the generated faces),

(12) the computer outlines the remaining object faces with solid lines, and

(13) the user accepts the faces of the object completing the face information for this object,

(14) when done, the computer indicates the number of faces generated.

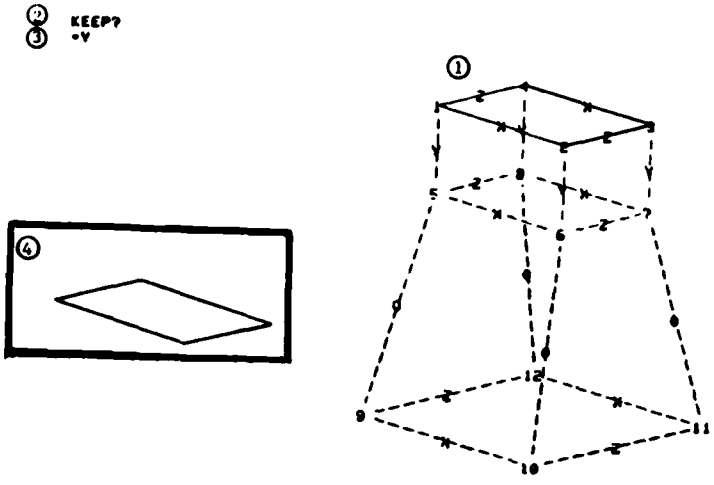


Figure 18. GENERATE FACES - Display of Face 1 (1-2-3-4-1)

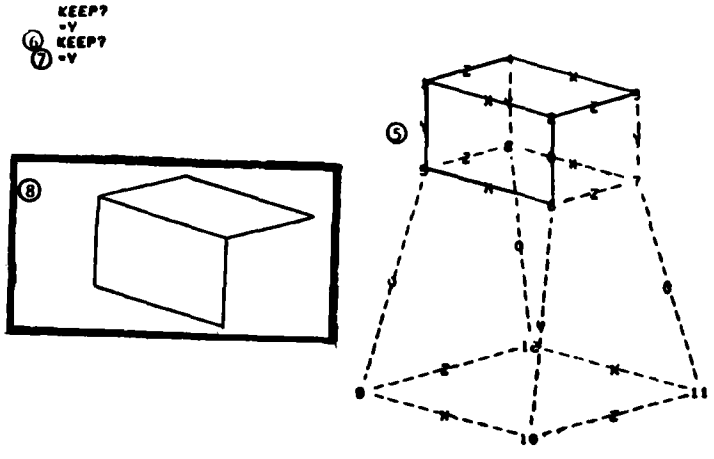


Figure 19. GENERATE FACES - Display of Face 2 (1-5-6-2-1)

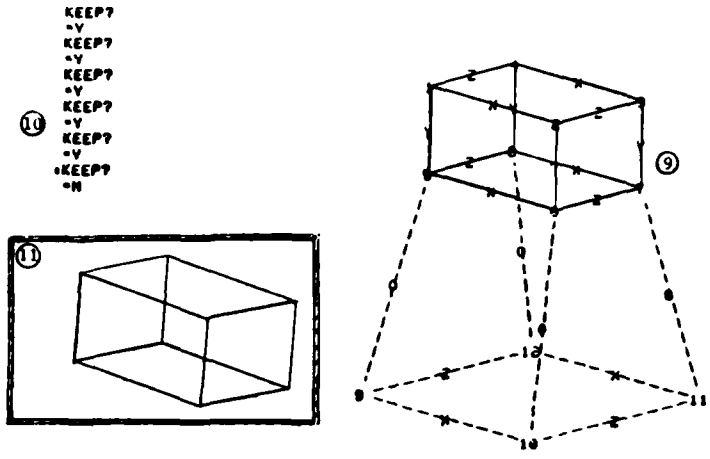


Figure 20. GENERATE FACES - Rejection of Face (5-6-7-8-5)

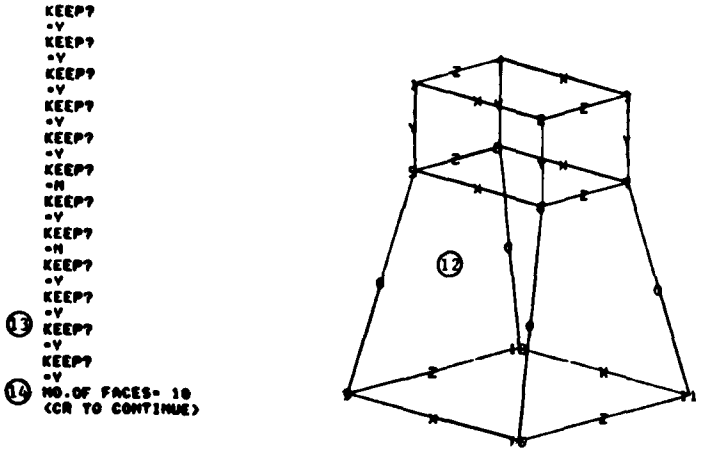


Figure 21. Completion of Face Generation

```
OBJECT    NAME=JUNK
COORDINATES
    1        0.500        7.000        3.000
    2        3.500        7.000        3.000
    3        3.500        7.000        1.000
    4        0.500        7.000        1.000
    5        0.500        5.000        3.000
    6        3.500        5.000        3.000
    7        3.500        5.000        1.000
    8        0.500        5.000        1.000
    9        0.000        0.000        4.000
   10        4.000        0.000        4.000
   11        4.000        0.000        0.000
   12        0.000        0.000        0.000
LINES
    1    2    1    X
    2    3    2    Z
    3    4    3    X
    4    1    4    Z
    5    5    1    Y
    6    5    6    X
    7    7    6    Z
    8    8    7    X
    9    5    8    Z
   10    4    8    Y
   11    3    7    Y
   12    2    6    Y
   13    9    5    0
   14   10    9    X
   15   11   10    Z
   16   12   11    X
   17    9   12    Z
   18    6   10    0
   19    7   11    0
   20    8   12    0
FACES
    1    5    1    2    3    4    1
    2    5    2    1    5    6    2
    3    5    3    2    6    7    3
    4    5    4    3    7    8    4
    5    5    1    4    8    5    1
    6    5    6    5    9   10    6
    7    5    7    6   10   11    7
    8    5    8    7   11   12    8
    9    5    5    8   12    9    5
   10    5   10    9   12   11   10
END
```

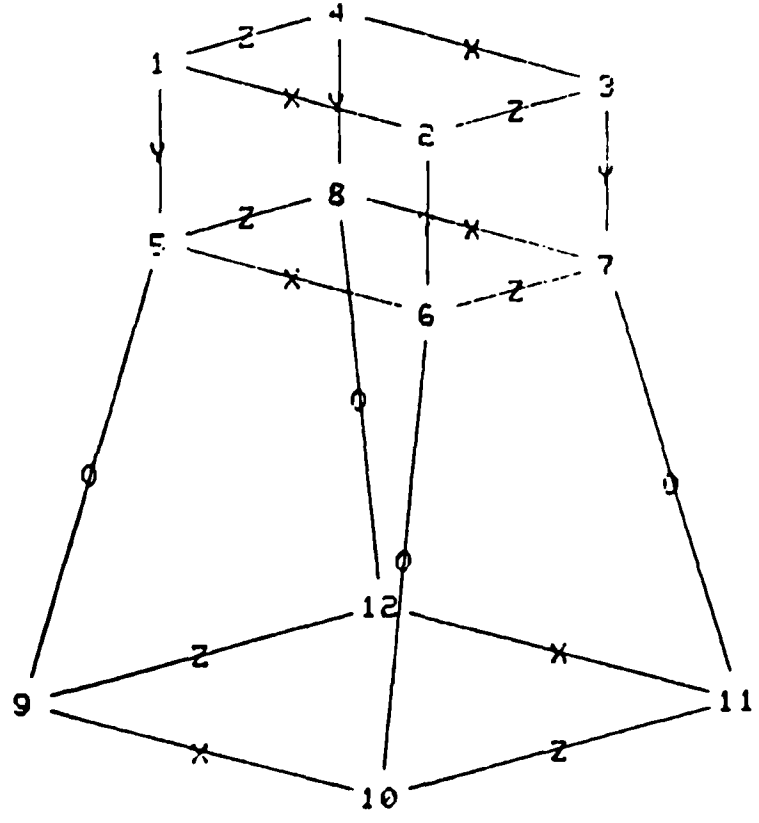


Figure 22. The Object File and Corresponding Display of the 3D Object

WATERWAYS EXPERIMENT STATION REPORTS PUBLISHED UNDER THE COMPUTER-AIDED STRUCTURAL ENGINEERING (CASE) PROJECT

	Title	Date
Technical Report K-78-1	List of Computer Programs for Computer-Aided Structural Engineering	Feb 1978
Instruction Report O-79-2	User's Guide: Computer Program with Interactive Graphics for Analysis of Plane Frame Structures (CFRAME)	Mar 1979
Technical Report K-80-1	Survey of Bridge-Oriented Design Software	Jan 1980
Technical Report K-80-2	Evaluation of Computer Programs for the Design/Analysis of Highway and Railway Bridges	Jan 1980
Instruction Report K-80-1	User's Guide: Computer Program for Design/Review of Curvilinear Conduits/Culverts (CURCON)	Feb 1980
Instruction Report K-80-3	A Three-Dimensional Finite Element Data Edit Program	Mar 1980
Instruction Report K-80-4	A Three-Dimensional Stability Analysis/Design Program (3DSAD) Report 1: General Geometry Module Report 3: General Analysis Module (CGAM)	 Jun 1980 Jun 1982
Instruction Report K-80-6	Basic User's Guide: Computer Program for Design and Analysis of Inverted-T Retaining Walls and Floodwalls (TWDA)	Dec 1980
Instruction Report K-80-7	User's Reference Manual: Computer Program for Design and Analysis of Inverted-T Retaining Walls and Floodwalls (TWDA)	Dec 1980
Technical Report K-80-4	Documentation of Finite Element Analyses Report 1: Longview Outlet Works Conduit Report 2: Anchored Wall Monolith, Bay Springs Lock	 Dec 1980 Dec 1980
Technical Report K-80-5	Basic Pile Group Behavior	Dec 1980
Instruction Report K-81-2	User's Guide: Computer Program for Design and Analysis of Sheet Pile Walls by Classical Methods (CSHTWAL) Report 1: Computational Processes Report 2: Interactive Graphics Options	 Feb 1981 Mar 1981
Instruction Report K-81-3	Validation Report: Computer Program for Design and Analysis of Inverted-T Retaining Walls and Floodwalls (TWDA)	Feb 1981
Instruction Report K-81-4	User's Guide: Computer Program for Design and Analysis of Cast-in-Place Tunnel Linings (NEWTUN)	Mar 1981
Instruction Report K-81-6	User's Guide: Computer Program for Optimum Nonlinear Dynamic Design of Reinforced Concrete Slabs Under Blast Loading (CBARCS)	Mar 1981
Instruction Report K-81-7	User's Guide: Computer Program for Design or Investigation of Orthogonal Culverts (CORTCUL)	Mar 1981
Instruction Report K-81-9	User's Guide: Computer Program for Three-Dimensional Analysis of Building Systems (CTABS80)	Aug 1981
Technical Report K-81-2	Theoretical Basis for CTABS80: A Computer Program for Three-Dimensional Analysis of Building Systems	Sep 1981
Instruction Report K-82-6	User's Guide: Computer Program for Analysis of Beam-Column Structures with Nonlinear Supports (CBEAMC)	Jun 1982
Instruction Report K-82-7	User's Guide: Computer Program for Bearing Capacity Analysis of Shallow Foundations (CBEAR)	Jun 1982
Instruction Report K-83-1	User's Guide: Computer Program With Interactive Graphics for Analysis of Plane Frame Structures (CFRAME)	Jan 1983
Instruction Report K-83-2	User's Guide: Computer Graphics Program for Generation of Engineering Geometry (SKETCH)	Jan 1983

END

FILMED

10-83

DTIC